

**This Page is Inserted by IFW Indexing and Scanning  
Operations and is not part of the Official Record**

**BEST AVAILABLE IMAGES**

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ **BLACK BORDERS**
- ☐ **IMAGE CUT OFF AT TOP, BOTTOM OR SIDES**
- ☐ **FADED TEXT OR DRAWING**
- ☐ **BLURRED OR ILLEGIBLE TEXT OR DRAWING**
- ☐ **SKEWED/SLANTED IMAGES**
- ☐ **COLOR OR BLACK AND WHITE PHOTOGRAPHS**
- ☐ **GRAY SCALE DOCUMENTS**
- ☐ **LINES OR MARKS ON ORIGINAL DOCUMENT**
- ☐ **REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY**
- ☐ **OTHER:** \_\_\_\_\_

**IMAGES ARE BEST AVAILABLE COPY.**

**As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.**

224



# UNITED STATES PATENT AND TRADEMARK OFFICE

UNITED STATES DEPARTMENT OF COMMERCE  
United States Patent and Trademark Office  
Address: COMMISSIONER FOR PATENTS  
P.O. Box 1450  
Alexandria, Virginia 22313-1450  
www.uspto.gov

APPLICATION NO.	FILING DATE	FIRST NAMED INVENTOR	ATTORNEY DOCKET NO.	CONFIRMATION NO.
09/703,449	10/31/2000	Stepan Sokolov	SUN1P814/P5417	1902
22434	7590	08/27/2004	EXAMINER	
BEYER WEAVER & THOMAS LLP			KENDALL, CHUCK O	
P.O. BOX 778			ART UNIT	
BERKELEY, CA 94704-0778			PAPER NUMBER	

2122

DATE MAILED: 08/27/2004

Please find below and/or attached an Office communication concerning this application or proceeding.

**Office Action Summary**

Application No.

09/703,449

Applicant(s)

SOKOLOV ET AL.

Examiner

Chuck Kendall

Art Unit

2122

-- The MAILING DATE of this communication appears on the cover sheet with the correspondence address --

**Period for Reply**

A SHORTENED STATUTORY PERIOD FOR REPLY IS SET TO EXPIRE 3 MONTH(S) FROM THE MAILING DATE OF THIS COMMUNICATION.

- Extensions of time may be available under the provisions of 37 CFR 1.136(a). In no event, however, may a reply be timely filed after SIX (6) MONTHS from the mailing date of this communication.
- If the period for reply specified above is less than thirty (30) days, a reply within the statutory minimum of thirty (30) days will be considered timely.
- If NO period for reply is specified above, the maximum statutory period will apply and will expire SIX (6) MONTHS from the mailing date of this communication.
- Failure to reply within the set or extended period for reply will, by statute, cause the application to become ABANDONED (35 U.S.C. § 133). Any reply received by the Office later than three months after the mailing date of this communication, even if timely filed, may reduce any earned patent term adjustment. See 37 CFR 1.704(b).

**Status**

- 1) ☒ Responsive to communication(s) filed on 03 August 2004.
- 2a) ☐ This action is **FINAL**. 2b) ☒ This action is non-final.
- 3) ☐ Since this application is in condition for allowance except for formal matters, prosecution as to the merits is closed in accordance with the practice under *Ex parte Quayle*, 1935 C.D. 11, 453 O.G. 213.

**Disposition of Claims**

- 4) ☒ Claim(s) 1-30 is/are pending in the application.
- 4a) Of the above claim(s) \_\_\_\_\_ is/are withdrawn from consideration.
- 5) ☐ Claim(s) \_\_\_\_\_ is/are allowed.
- 6) ☒ Claim(s) 1-30 is/are rejected.
- 7) ☐ Claim(s) \_\_\_\_\_ is/are objected to.
- 8) ☐ Claim(s) \_\_\_\_\_ are subject to restriction and/or election requirement.

**Application Papers**

- 9) ☐ The specification is objected to by the Examiner.
- 10) ☐ The drawing(s) filed on \_\_\_\_\_ is/are: a) ☐ accepted or b) ☐ objected to by the Examiner.  
Applicant may not request that any objection to the drawing(s) be held in abeyance. See 37 CFR 1.85(a).  
Replacement drawing sheet(s) including the correction is required if the drawing(s) is objected to. See 37 CFR 1.121(d).
- 11) ☐ The oath or declaration is objected to by the Examiner. Note the attached Office Action or form PTO-152.

**Priority under 35 U.S.C. § 119**

- 12) ☐ Acknowledgment is made of a claim for foreign priority under 35 U.S.C. § 119(a)-(d) or (f).
- a) ☐ All b) ☐ Some \* c) ☐ None of:
1. ☐ Certified copies of the priority documents have been received.
  2. ☐ Certified copies of the priority documents have been received in Application No. \_\_\_\_\_.
  3. ☐ Copies of the certified copies of the priority documents have been received in this National Stage application from the International Bureau (PCT Rule 17.2(a)).

\* See the attached detailed Office action for a list of the certified copies not received.

**Attachment(s)**

- |  |   |
|--|---|
| 1) <input checked="" type="checkbox"/> Notice of References Cited (PTO-892) *  | 4) <input type="checkbox"/> Interview Summary (PTO-413)<br>Paper No(s)/Mail Date. _____ |
| 2) <input type="checkbox"/> Notice of Draftsperson's Patent Drawing Review (PTO-948)                                   | 5) <input type="checkbox"/> Notice of Informal Patent Application (PTO-152)             |
| 3) <input type="checkbox"/> Information Disclosure Statement(s) (PTO-1449 or PTO/SB/08)<br>Paper No(s)/Mail Date _____ | 6) <input type="checkbox"/> Other: _____  |

### DETAILED ACTION

1. This action is in response to the application filed 8/03/04.
2. Claims 1 – 30 are pending.

### Claim Rejections - 35 USC § 112

3. The following is a quotation of the second paragraph of 35 U.S.C. 112:

The specification shall conclude with one or more claims particularly pointing out and distinctly claiming the subject matter which the applicant regards as his invention.

4. Claims 3, 5, 7 – 10, 13 – 15 & 18 – 28, contains the trademark/trade name Java TM. Where a trademark or trade name is used in a claim as a limitation to identify or describe a particular material or product, the claim does not comply with the requirements of 35 U.S.C. 112, second paragraph. See *Ex parte Simpson*, 218 USPQ 1020 (Bd. App. 1982). The claim scope is uncertain since the trademark or trade name cannot be used properly to identify any particular material or product. A trademark or trade name is used to identify a source of goods, and not the goods themselves. Thus, a trademark or trade name does not identify or describe the goods associated with the trademark or trade name. In the present case, the trademark/trade name is used to identify/describe a load constant command and, accordingly, the identification/description is indefinite.

**Claim Rejections - 35 USC § 103**

5. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

6. Claims 1 – 9, 11 – 21, 29 and 30 are rejected under 35 U.S.C. 103(a) as being unpatentable over Augusteijn et al. USPN 6,292,883 B1 (prior art of record), in view of Grove USPN 6,205,578 B1 (prior art made of record).

---

Regarding claims 1 & 21, Augusteijn discloses a method of creating data structures, an object oriented programming environment (1:44 – 46, JAVA TM) suitable for use by a virtual machine to execute computer instructions. Although, Augusteijn doesn't explicitly disclose converting a stream of virtual machine commands and data associated with the virtual machine commands into a pair of streams for use in the virtual machine, the pair of streams including a code stream and a data stream, wherein the code stream includes the virtual machine commands, and wherein the data stream includes the data associated with the commands in the code stream, Augusteijn does however, disclose converting virtual machine instructions (commands) into native instructions for a program specific virtual machine 2: 62, which includes commands and associated data (12: 1 – 5 and 8 – 15, see selection data, conversion data and virtual instructions). Grove discloses an interpreter which determines bytecode from a bytecode stream, (i.e. opcode) as well as additional data from the bytecode (e.g. operands or other data) 8:3 – 8. Grove further discloses that during the execution of a java application or program a thread is either executing bytecodes or native methods (i.e., methods implemented in a language other than java...), see 5: 65 – 6:2. Therefore it would have been obvious to one of ordinary skill in the art at the time the invention

was made to combine Augusteijn and Grove because, performing conversions utilizing bytecodes would make the system more portable and platform independent.

Regarding claims 2 & 12, Grove further discloses a method as recited and previously discussed in claim 1, wherein the code stream includes only commands and the data stream includes only the data associated with the commands in the code stream (Grove, 6: 54 – 60).

Regarding claims 3,13, &18 a method as recited in claim 1, wherein the stream that is to be converted is a JAVA TM compliant bytecode stream, the code stream is a JAVA TM bytecode code stream that includes JAVA TM commands, and the data stream is a JAVA TM bytecode data stream that includes the data associated with the JAVA TM commands in the JAVA TM bytecode code stream (Augusteijn, 13:28 – 30, note JAVA TM bytecode is the same as a virtual machine instruction as stated in prior art, and refer to, 7:3 –10, 23 – 30, also see Grove 8: 1 – 15).

Regarding claim 4, a method as recited in claim 1, wherein said converting of said stream comprises:

writing a representation of a first command associated with a first instruction into a code entry of the code stream (Augusteijn, 2:63 – 3: 10, see table and microcode and storing for writing, also see 12:9, for storing selection data);

determining whether the first command has data associated with it (Augusteijn 12: 13 – 15, for associated see conversion data indicated by selection data); and

writing a representation of the associated data or a reference to a representation of the data associated with the first command into a first data entry of the data stream when the command has associated data (Augusteijn,12: 1 - 15, see conversion table for data entry or data stream).

Regarding claim 5, a method as recited in claim 4, wherein the stream that is to be converted is a JAVA TM bytecode stream, the code stream is a JAVA TM bytecode code stream that includes JAVA TM commands, and the data stream is a JAVA TM bytecode data stream that includes the data associated with the JAVA TM commands in the JAVA TM bytecode code stream (10:30 – 35, for writing data see “stored” also see Grove 8: 1 – 15).

Regarding claim 6, a method as recited in claim 4, wherein said method further comprises:

not providing a data entry in the data stream for the command when the command does not have data associated with it (Augusteijn, 10:23 – 27, see “ may use autonomous conversion” or “ shared logic”).

Regarding claim 7, a method as recited in claim 6, wherein said method further comprises:

writing a representation of second command associated with another instruction into a second code entry of the code stream (Augusteijn, 10:25 – 30, see for each of the virtual machines and fig. 3 shows 2 VM's, 332 and 336);

determining whether the second command has data associated with it; and writing a representation of the associated data or a reference to a representation of the data associated with the second command into a second data entry of the data stream when the command has associated data (Augusteijn, 10:20 – 35 ).

Regarding claim 8, a method as recited in claim 7, wherein the stream that is to be converted is a JAVA TM bytecode stream, the code stream is a JAVA TM bytecode code stream that includes JAVA TM commands, and the data stream is a JAVA TM bytecode data stream that includes the data associated with the JAVA TM commands in the JAVA TM bytecode code stream (Augusteijn, 7:23 – 30).

Regarding claims 9 & 14, a method as recited in claim 8, wherein the command and data entries can each include a number of bytecodes in the JAVA TM bytecode stream, wherein the number of bytecodes is an integer, and wherein each byte code can be one or more bytes (Augusteijn, 13:25 – 30).

Regarding claim 11, in an object oriented programming environment a computer readable medium including computer program code generating computer executable commands and data associated with the computer executable commands suitable for use by a virtual machine and comprising:

computer program for receiving a first stream of virtual machine instructions that include virtual machine commands and data associated with the commands (Augusteijn, 11: 62 – 67);

Art Unit: 2122

computer program code for generating, from the first stream, a code stream having one or more virtual machine commands (Augusteijn, 7:1 – 15); and

computer program code for generating, from the first stream, a data stream having data associated with the one or more virtual machine commands (Augusteijn, 9:13 – 37, see selection data and virtual machine instruction).

Regarding claim 15, a data structure as recited in claim 14, wherein the JAVA<sup>TM</sup> commands can be a load constant command, an invoke method command, a jump command, an instantiation command, or a get/put field command (Augusteijn, 6:30 – 45, see pointing by pointer / jump command, class loader and retrieve field).

Regarding claim 16, see claim 1 for reasoning.

Regarding claim 17, a method as recited in claim 16, wherein the method further comprises:

---

updating a pointer to the command stream (Augusteijn, 11:20 – 25 , see change of instruction pointer); and

updating a pointer to the data stream (Augusteijn, 10:30 – 40).

Regarding claim 19, see claim 14 for reasoning.

Regarding claim 20, is the method claim corresponding to claim 15 and is rejected using the same rationale, as in claim 15.

Regarding claim 29, Augusteijn discloses a method of executing a virtual machine instruction on a virtual machine, the method comprising:

wherein the virtual machine instruction has a code portion and one or more data portions the converting the virtual machine instruction into a pair of streams, the pair of representation of the code portion of the virtual machine instruction, and wherein the data stream provides data that is needed to execute the code stream (Augusteijn, 7:3 – 10 , 23 – 30, also see 12: 1 – 5 and 8 – 15, see selection data, conversion data and virtual instructions);

reading the code portion from the codes stream (Augusteijn, 11:56 – 59);

directly accessing the data from the code stream (Augusteijn, 7:20 – 25); and



executing the virtual machine instruction using the data directly accessed from the code portion (Augusteijn, 8: 13 – 20). However, Augusteijn doesn't explicitly disclose

reading a virtual machine instruction from a first stream (Augusteijn, 11:56 – 59);

Regarding claim 30, a method as recited in claim 29, wherein said converting the virtual machine instruction is performed at load time when a class file that includes the virtual machine instruction is loaded in the virtual machine (Augusteijn, FIG.3, 342 and 346, also see associated text).

### **Claim Rejections - 35 USC § 103**

7. The following is a quotation of 35 U.S.C. 103(a) which forms the basis for all obviousness rejections set forth in this Office action:

(a) A patent may not be obtained though the invention is not identically disclosed or described as set forth in section 102 of this title, if the differences between the subject matter sought to be patented and the prior art are such that the subject matter as a whole would have been obvious at the time the invention was made to a person having ordinary skill in the art to which said subject matter pertains. Patentability shall not be negated by the manner in which the invention was made.

8. Claim 10 is rejected under 35 U.S.C. 103(a) as being unpatentable over Augusteijn et al. USPN 6,292,883 B1 (prior art of record), in view of Grove USPN 6,205,578 B1 (prior art made of record) as applied in claim 9, in view of Wahbe et al. USPN 6,151,618 (hereinafter Wahbe) art of record.

Regarding claim 10, Augusteijn and Grove discloses all the claimed limitations as applied in claim 9 above. The combination of Augusteijn and Grove does not explicitly disclose wherein the first and second entries of the code stream are adjacent to each other. However, Wahbe does disclose this feature (17:33 – 37). Therefore, it would have been obvious to one of ordinary skill in the art at the time the invention was made to combine Augusteijn and Grove with Wahbe because, associating or combining

adjacent entries of code, "This ensures that the operand specialization technique will not compete with the opcode combination technique by further specializing an instruction before the combiner has a chance to consider a less-specialized version" (Wahbe, 17:50 – 55).

9. Claims 22 – 28 are rejected under 35 U.S.C. 103(a) as being unpatentable over Augusteijn et al. USPN 6,292,883 B1 (prior art of record), in view of Grove USPN 6,205,578 B1 (prior art made of record) as applied in claim 21, in view of Toutonghi et al. USPN 5,292,720 (hereinafter Toutonghi) art of record.

Regarding claim 22, Augusteijn and Grove discloses all the claimed limitations as applied in claim 21, above. The combination of Augusteijn and Grove doesn't explicitly reading and processing the associated data from a Constant Pool when the command has an associated data. However, Toutonghi does disclose this feature (6:37 – 40). Therefore, it would have been obvious to one of ordinary skill in the art at the invention was made to combine Augusteijn and Grove with Toutonghi because, associating static data or constant data in a JAVA TM environment using a constant pool, would make associating the variables or data more efficient.

Regarding claim 23, a method as recited in claim 22, wherein said processing operates to determine a constant value associated with a JAVA TM Load Constant command (Toutonghi, 5:50-55, see verifier which checks form of the bytecodes, also see fig.2, 114, 100).

Regarding claim 24, a method as recited in claim 22, wherein said processing operates to determine a reference to a method invocation cell that includes information relating to a JAVA TM invoke method command (Toutonghi, fig.2, 112).

Regarding claim 25, a method as recited in claim 22, wherein said processing operates to determine the code stream offset and data stream offset associated with a JAVA TM jump command (Toutonghi, 3:10 – 15).

Regarding claim 26, a method as recited in claim 22, wherein said processing operates to process a Constant Pool associated with a JAVA TM instantiation command (Toutonghi, 3:5 – 15).

Regarding claim 27, a method as recited in claim 22, wherein said processing operates to process a Constant Pool associated with a JAVA TM Get/Put field command (Toutonghi, 6:35 – 46 , for retrieve data field).

Regarding claim 28, a method as recited in claim 22, wherein said processing operates to process data associated with a JAVA TM load constant command, a JAVA TM invoke method command, a JAVA TM jump command, a JAVA TM instantiation command, or a JAVA TM get/put field command (Toutonghi, 3:1 – 15, see method, table pointer (jump command), instance, see 6:35 – 46, for retrieve data field).

---

### ***Response to Arguments***

10. Applicant's arguments filed 08/03/2004 with regards to claims 1 – 30 have been fully considered but are moot in view of new grounds of rejection.

### **Correspondence Information**

11. Any inquiry concerning this communication or earlier communications from the examiner should be directed to Chuck Kendall whose telephone number is 703-3086608. The examiner can normally be reached on 10:00 am - 6:30pm.

If attempts to reach the examiner by telephone are unsuccessful, the examiner's supervisor, Tuan Dam can be reached on 703-3054552. The fax phone number for the organization where this application or proceeding is assigned is 703-872-9306.

Art Unit: 2122

Information regarding the status of an application may be obtained from the Patent Application Information Retrieval (PAIR) system. Status information for published applications may be obtained from either Private PAIR or Public PAIR. Status information for unpublished applications is available through Private PAIR only. For more information about the PAIR system, see <http://pair-direct.uspto.gov>. Should you have questions on access to the Private PAIR system, contact the Electronic Business Center (EBC) at 866-217-9197 (toll-free).

\*\*\*

---



**ANTHONY NGUYEN-BA**  
**PRIMARY EXAMINER**